APPARATUS AND METHOD FOR DE-INTERLEAVING THE INTERLEAVED DATA IN A CODED ORTHOGONAL FREQUENCY DIVISION MULTIPLEXING RECEIVER

FIELD OF THE INVENTION

[0001] The present invention generally relates to the field of processing the interleaved data in coded orthogonal frequency division multiplexing (COFDM) communication systems, such as a digital audio broadcasting (DAB) system, and more particularly to an apparatus and method for de-interleaving the interleaved data in a COFDM receiver.

BACKGROUND OF THE INVENTION

[0002] In communication systems, data are often distorted by channel impairments (e.g., fading, multipath) during transmission. This may cause the so-called burst error, i.e. successively erroneous data. An interleaver/de-interleaver is generally used to overcome the effect of burst errors. Specifically, the interleaver interleaves successive data into non-successive data for transmission. If burst of errors occurs in the interleaved data, the de-interleaver in the receiver will de-interleave the successively erroneous data into non-successive data. Accordingly, the de-interleaved data would not have large amount of burst errors. This thus prevents the channel decoder in the receiver from overloading its error correction capability for correcting the large amount of successively erroneous data.

[0003] For COFDM systems such as a DAB system, time interleaving is used against the burst errors caused by time-varying channel fading, and frequency interleaving and block interleaving are used against the burst errors caused by multipath propagations. In

DAB system, the transmitter performs time interleaving, block interleaving, and frequency interleaving in a sequential order. In the receiver, the complementary operations, frequency de-interleaving, block de-interleaving and time de-interleaving, must be performed in a reverse order, so that the de-interleaved data have the same order as the original ones.

[0004] FIG. 1a is a block diagram illustrating the interleaving process in a DAB transmitter. As can be seen from FIG. 1a, the interleaving process includes three procedures: time interleaving, block interleaving and frequency interleaving. First, a time interleaver 101, which is a convolutional interleaver, interleaves the input data bit stream according to a cyclic repeated delay pattern to generate a time interleaved bit stream 102. This time interleaved bit stream 102 is fed to a block interleaver 103. The block interleaver 103 maps every two data bits in the time interleaved bit stream 102 into one symbol in a manner that is illustrated by the followings. Two blocks of successive data bits are shown in the time interleaved bit stream 102: a block comprising bits b[0]...b[N-1] and another block comprising bits b[N]...b[2N-1], where N is the number of used sub-carriers. Each bit in the block b[0]...b[N-1] is paired with the corresponding bit in the block b[N]...b[2N-1] to form a mapped symbol (b[j], b[j+N]), $j \in [0, N-1]$, where b[j] and b[j+N] represent the in-phase (I) component and quadrature-phase (Q) component of the mapped symbol (b[j], b[j+N]). The mapped symbols (b[0], b[N]), ..., (b[N-1], b[2N-1]) are then interleaved to non-successive sub-carriers via a frequency interleaver 105. Finally, the OFDM modulator 107 generates modulated samples by performing differential quarternary phase shift keying (QPSK) modulation and multicarrier modulation.

[0005] FIG. 1b is a block diagram illustrating the de-interleaving process in a DAB receiver. It shows that the de-interleaving process is performed in a reverse order with respect to that in FIG. 1a. Referring to FIG. 1b, the OFDM demodulator 109 generates demodulated samples by performing multicarrier demodulation and differential QPSK demodulation. Each demodulated sample, denoted by $(\hat{b}[j],\hat{b}[j+N])$, j∈[0, N-1], is typically represented as a $2m$-bit metric, where the most significant m bits and least significant m bits correspond to the I and Q metric of the demodulated sample, respectively. Feeding the demoduled samples into the frequency de-interleaver 111, block de-interleaver 113, and time de-interleaver 117 gives rise to the de-interleaved data in its original pre-interleaved form. This de-interleaved data (I and Q metric) are then presented to the channel decoder for channel decoding.

[0006] De-interleaving process generally needs considerable sizes of storage units. A conventional de-interleaving approach with two stage operations is known to involve a buffer for performing frequency de-interleaving and block de-interleaving in one stage and a memory for performing time de-interleaving in another stage. Note that the OFDM demodulator 109 is typically implemented in a manner of serial-in-serial-out. This implies that after frequency de-interleaving and block de-interleaving, the first output sample from the OFDM demodulator 109 may not correspond to the first sample to be fed to the time de-interleaver 117, and so as for the other samples. Accordingly, a buffer is required for the conventional two-stage approach to temporarily store the entire frequency de-interleaved and block de-interleaved data of an OFDM symbol.

[0007] Notice that data are written into the above-mentioned buffer in a non-sequential order due to the frequency de-interleaving and block de-interleaving, but they

are typically read out from the buffer in a sequential order for further time de-interleaving. As such, writing data into the buffer may overwrite previously stored data that have not been read out yet. A typical approach to solve this problem is to read data from the buffer in the guard interval of an OFDM symbol while writing data into the buffer in the useful interval of the following OFDM symbol. This approach, however, requires a higher rate of reading data from the buffer as well as a higher rate of performing the following time de-interleaving operations, since the guard interval is generally much less than the useful interval.

[0008] The conventional two-stage approach, thus has two main drawbacks: (1) lager circuitry due to the use of frequency and block de-interleaving buffer and (2) higher power consumption due to higher processing rate for de-interleaving the interleaved data.

SUMMARY OF THE INVENTION

[0009] The present invention has been made to overcome the above-mentioned drawbacks of conventional de-interleaving process in a COFDM receiver. An object of the present invention is to provide a cost- and power-effective de-interleaving apparatus in a COFDM receiver. Specifically, the apparatus of this invention comprises a de-interleaving memory and a controller. The controller generates memory read/write control signals as well as memory access addresses for the frequency de-interleaving, block de-interleaving and time de-interleaving process. The demodulated samples are directly stored into and read out from the de-interleaving memory according to the read/write controls and the access addresses generated by the controller with no need of any additional buffer during the de-interleaving process.

[0010] Another object of the invention is to provide a method for de-interleaving the interleaved data in a COFDM receiver. In the method, frequency de-interleaving, block de-interleaving and time de-interleaving are performed in a manner of one-stage approach. Accordingly, the method of this invention comprises the steps of (a) generating the correct access addresses of the de-interleaving memory for the interleaved data; and (b) de-interleaving the interleaved data in a manner of performing frequency de-interleaving, block de-interleaving and time de-interleaving with only the de-interleaving memory.

[0011] The strategy for generating the access addresses of the de-interleaving memory is to first define the memory size required for de-interleaving an OFDM symbol and then generate pointers including segment pointer, group pointer, sub-group pointer, and cell pointer to allocate the correct access addresses of the memory.

[0012] With the de-interleaving memory, each pair of (b[j], b[j+N]), $j \in [0, N-1]$, from the demodulator output can be stored as a basic memory cell. The block de-interleaving can then be effectively performed by reading out either the most or least significant m bits from the de-interleaving memory.

[0013] An advantage of the invention is relatively low memory cost. Only the de-interleaving memory is required to perform the de-interleaving process in a COFDM receiver. No additional buffer is needed to proceed the frequency de-interleaving and block de-interleaving for the interleaved data.

[0014] It is yet another advantage of the invention to further reduce the cost of the de-interleaving circuitry by directly using the de-interleaving memory combined with the access addresses of the memory. Accordingly, the controller is used to generate the

correct access addresses of the memory and controls the read and write operations to the memory.

[0015] The foregoing and other objects, features, aspects and advantages of the present invention will become better understood from a careful reading of the detailed description provided herein below with appropriate reference to the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

[0016] The present invention can be understood in more detail by reading the subsequent detailed description in conjunction with the examples and references made to the accompanying drawings, wherein:

FIG. 1a is a block diagram illustrating the interleaving process in a DAB transmitter;

FIG. 1b is a block diagram illustrating the de-interleaving process in a DAB receiver;

FIG. 2 shows the delay pattern for each bit in a CIF to perform the time interleaving and time de-interleaving;

FIG. 3 is a block diagram of an embodiment according to the invention for de-interleaving the interleaved data using a de-interleaving memory in a COFDM receiver;

FIG. 4 is a diagram illustrating the four pointers, segment pointer, group pointer, sub-group pointer and cell pointer; and

FIG. 5 shows an example of the memory read/write address generator for the controller in the embodiment of FIG. 3.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

[0017] It is known that the interleaved data stream is transmitted according to an OFDM modulation technique in a DAB system. Each one of the common interleaved frames (CIF) in a DAB system contains 55296 bits. The time interleaving is performed by a unit of 16 bits according to a cyclic repeated delay pattern, and each bit in one cycle corresponds to a certain number of delay units that is required to perform the time interleaving. This is further shown in FIG. 2, where a small blank square represents a delay unit (Δ). In DAB, a delay unit equals to 24 milliseconds, and it is the duration of a CIF too. In a receiver, the time de-interleaver thus needs a complementary number of delay units for each bit in a cycle to perform the time de-interleaving. The left-hand side and right-hand side of FIG. 2 show the number of delay units for each bit in a CIF to perform the time interleaving and time de-interleaving, respectively.

[0018] The gist of the present invention is using a de-interleaving memory combined with memory address computation for de-interleaving the interleaved data in a COFDM receiver. FIG. 3 shows a block diagram of the apparatus for de-interleaving the interleaved data using a de-interleaving memory in a COFDM receiver according to this invention. The apparatus comprises a controller 301, and a de-interleaving memory 303.

[0019] Referring to FIG. 3, when a demodulator provides demodulated samples usually in a manner of serial-in-serial-out, the de-interleaving memory 303 is used for de-interleaving the demodulated samples. The controller 301 generates the correct access addresses of the de-interleaving memory 303 for the data to be de-interleaved, and controls the read and write operations to the de-interleaving memory 303. In order to generate the correct access addresses, the controller 301 further comprises an address

generator having a plurality of base address generators, and a read/write control circuitry. These base address generators include a segment base address generator, a group base address generator, a sub-group base address generator, and a cell base address generator. A base address generator includes a pointer that indicates the starting address of a segment, a group or a sub-group in the de-interleaving memory 303, or the offset address of a cell in a sub-group. FIG. 4 is a diagram illustrating the four pointers. How to create these pointers and calculate the correct read/write address of the de-interleaving memory 303 will be discussed further in the description.

[0020] First, a segment is defined as the required memory capacity in the de-interleaving memory 303 for de-interleaving an OFDM symbol, and the segment size needs to be determined in the design of this invention. Mode I in Table 1 is used to illustrate on how to determine the segment size. As mentioned before, a CIF in a DAB system contains 55296 bits. For mode I, each OFDM symbol can transmit 3072 bits. In other words, each CIF includes 18 (=55296/3072) OFDM symbols. Each OFDM symbol can be further divided into 96 (=3072/16/2) groups, i.e., 96 cycles of repeated delay pattern.

[0021] Referring back to FIG. 2, the total delay units for each bit to perform the time interleaving and de-interleaving equal 16, i.e. the sum of the delay units in both sides for each row equals 16. In the receiver, a group size is determined by the total delay units in a cycle to perform the time de-interleaving. Therefore, the group size to perform the time de-interleaving in the receiver for mode I can be calculated as 16+8+12+4+14+6+10+2+15+6+11+3+5+9+1, that is the sum of the integers from 1 to 16, and equal to 136. Accordingly, a segment in mode I has 13,056 (=136*96) memory cells.

Table 1 shows the correspondence between the segment size and the number of total OFDM symbols in a CIF for four different modes, mode I ~ mode IV.

[0022] Next, a sub-group and a cell are defined, and each corresponding pointer needs to be determined so that the correct write address of the de-interleaving memory 303 can be calculated for the demodulated samples. In the invention, a sub-group is defined as a row in a group and a cell is defined as a memory cell in a row. Accordingly, the segment pointer points to the starting address of the current segment, the group pointer points to the starting address of the current group after the segment pointer has been determined, the sub-group pointer points to the current row after the group pointer has been determined, and the cell pointer points to the current cell after the sub-group pointer has been determined.

[0023] Consequently, the memory location for writing data after demodulation into the de-interleaving memory 303 is calculated as

$$Write\_address = segment\_pointe\_wr + group\_pointer\_wr + sub\text{-}group\_pointer\_wr + cell\_pointer\_wr. \quad (1)$$

In the following, how to calculate each pointer is further described.

[0024] For a DAB receiver, the segment size is given in Table 1. The segment pointer for writing data into the de-interleaving memory 303 is calculated as

$$segment\_pointer\_wr = ofdm\_symbol\_no * segment\_size, \quad (2)$$

where *ofdm_symbol_no* denotes the OFDM symbol number currently being proceeded.

[0025] To calculate the *group_pointer_wr*, an index *deitlvfreq_no* is defined as a frequency index after frequency de-interleaving. It can be obtained via referring to a look

up table (LUT) performed an index re-numbering operation on the output sub-carrier index of the demodulator. As known before, the time de-interleaving is performed by a unit of 16 bits according to a cyclic repeated delay pattern in a CIF. Then, *group_pointer_wr* can be calculated as

$$group\_pointer\_wr = Q(deitlvfreq\_no/16) * group\_size, \quad (3)$$

where $Q(x/y)$ denotes the quotient of x/y, and *group_size* denotes the required memory capacity in a group. For mode I, *group_size* equals 136.

[0026] After having the group pointer, the starting address for the current group is known. The sub-group pointer will be calculated to point to the row in the current group the data should be written into. This can be computed by taking the remainder of *deitlvfreq_no/16*. For calculating address for the current row in the group, the offset from the starting address of the current row to the starting address of the current group can also be found out by referring to a relatively small look up table (LUT) shown in Table 2, where Bit # relates to a row number in a group. For example, if the value of *deitlvfreq_no(mod 16)* is 3, then Bit # is 3. This means that it corresponds to the 4th row in a group, and the offset from the 4th row to the starting address of the group is 36. If Bit # is 15, then it relates to the 16th row and the offset 135. Therefore, *sub-group_pointer_wr* can be described as

$$sub\text{-}group\_pointe\_wr = LUT(deitlvfreq\_no(mod\ 16)). \quad (4)$$

[0027] After the pointers of segment, group and sub-group have been calculated, the correct write address can be determined by further computing the cell pointer. Because each sub-group requires a cell pointer to indicate which cell the data should be written

into, 16 cell pointers are required for 16 sub-groups. Each cell pointer is in the range from 0 to the number of delay units of its corresponding sub-group. For example, the number of delay units for sub-group 2 is 12, i.e. there are 12 cells in sub-group 2. Then cell pointer is in the range from 0 to 11 in a cyclic repeated pattern. It should be noted that each of the number of delay units for its corresponding sub-group is different for time de-interleaving; besides, each cell pointer automatically increases by one for every CIF duration, and goes on cyclic counting.

[0028] Finally, the write address is determined by summing up the four pointers, *segment_pointe_wr*, *group_pointer_wr*, *sub-group_pointer_wr* and *cell_pointer_wr*. The write address can be rewritten in a more specific way as follows.

$$Write\_address = ofdm\_symbol\_no * segment\_size + Q(deitlvfreq\_no/16) * group\_size + LUT(deitlvfreq\_no(mod\ 16)) + cell\_pointer\_wr \quad (5)$$

[0029] After having been delayed the time period of 16 CIFs, data can be read from the de-interleaving memory 303. When data is to be read from the de-interleaving memory 303, the four pointers, *segment_pointer_rd*, *group_pointer_rd*, *sub-group_pointer_rd* and *cell_pointer_rd*, are used to determine the read address of the data. While, it should be noted that the data to be written cannot overwrite the previous data which have not been read from the memory yet. Accordingly, during the time period of the data in the n^{th} segment being written into the memory, the data in the $(n+1)^{th}$ segment is being read from the memory.

[0030] More specifically, the index *deitlvfreq_no* for calculating the group pointer and sub-group pointer can be replaced by a read counter, *read_counter*, which counts

from 0 to N-1 and restarts from 0, where N is the number of used sub-carriers. For mode I, the read counter counts from 0 to 1535 for computing the memory read addresses. In order to prevent the stored data from being overwritten by new data, the value of *cell_pointer_rd* is the same as that of *cell_pointer_wr* and the read counter has two cyclic countings for an OFDM symbol. In other words, there are two rounds of reading during a period of an OFDM symbol. During the first round, the I metric (most significant *m* bits) in the location pointed by the read address are read from the memory. During the second round, the Q metric (least significant *m* bits) in the location pointed by the read address are read from the memory. This completes the de-interleaving process of frequency de-interleaving, block de-interleaving, and time de-interleaving. Therefore, the read address can be calculated as

$$Read\_address = (ofdm\_symbol\_no+1) * segment\_size + Q(read\_counter/16) * group\_size + LUT(read\_counter(mod\ 16)) + cell\_pointer\_rd. \quad (6)$$

[0031] Compared with the output data rate of the demodulator, this invention is only at the same rate for writing data into memory and is at the double rate for reading data from memory. Therefore, an advantage of the invention is relatively low power consumption, whereas the conventional two-stage approach requires higher processing rate for performing de-interleaving operations.

[0032] Recall that each demodulated data is a 2m-bit metric in which the most significant m bits and least significant m bits represent I metric and Q metric, respectively. If the demodulated data in one OFDM symbol are stored in the de-

interleaver, then this de-interleaving memory requires 884736*m* (=55296 *16**m*) bits to store the data of 16 CIFs. In the design of the present invention, only the space of useful data of 16 CIFs is required for the de-interleaving memory 303 and this amounts to only 470016*m* (=13056*18*2*m*) bits for mode I. Therefore, another advantage of the invention is relatively low memory cost.

[0033] FIG. 5 is a block diagram illustrating memory read/write address generator for the controller 301 in the embodiment of FIG. 3 according to the invention. Two multiplexers 501 and 502 determine whether read or write address is to be generated. When read address is to be generated, the multiplexer 501 selects *ofdm_symbol_no*+1 as its output, and the multiplexer 502 selects *read_counter* as its output. When write address is to be generated, the multiplexer 501 selects *ofdm_symbol_no* as its output, and the multiplexer 502 selects *deitlvfreq_no* as its output.

[0034] Referring back to the equations for *Write_address* and *Read_address*, it can be seen that the computation for either *Write_address* or *Read_address* needs two multiplications, a division of performing quotient and modulus operations, a look up table, and a cyclic counting as well as a summation which sums up the resultant values from the two multiplications, the look up table, and the counting. In other words, the address generator in the embodiment of FIG. 5 may includes two multipliers 511 and 512, a divider 503, a quotient unit 504, a remainder unit 505, a look up table 506, a counter 507, and an adder 508. Note that in DAB system, the divider with divisor 16 can be easily implemented using shift registers since the divisor 16 is an integer power of two. Therefore, implementation of the address generator is relatively simple.

[0035] It is also notable that there is no any additional buffering between the

demodulator and the de-interleaving memory 303 according to the present invention. By using the de-interleaving memory 303 combined with memory address computation for the interleaved data, frequency de-interleaving, block de-interleaving, and time de-interleaving are performed in one-stage approach in the COFDM receiver. All digital I, Q components of the plurality of phase-modulated sub-carries in the received DAB signal can be saved in the same memory, i.e. the de-interleaving memory 303 of the invention.

[0036] Although the present invention has been described with reference to the preferred embodiments, it should be understood that the invention is not limited to the details described thereof. Various substitutions and modifications have been suggested in the foregoing description, and others will occur to those of ordinary skill in the art. Therefore, all such substitutions and modifications are intended to be embraced within the scope of the invention as defined in the appended claims.